



## A Weighted Fuzzy Time Series Forecasting Model

Ortiz-Arroyo, Daniel; Poulsen, Jens Runi

*Published in:*  
Indian Journal of Science and Technology

*DOI (link to publication from Publisher):*  
[10.17485/ijst/2018/v11i27/130708](https://doi.org/10.17485/ijst/2018/v11i27/130708)

*Creative Commons License*  
CC BY 4.0

*Publication date:*  
2018

*Document Version*  
Publisher's PDF, also known as Version of record

[Link to publication from Aalborg University](#)

*Citation for published version (APA):*  
Ortiz-Arroyo, D., & Poulsen, J. R. (2018). A Weighted Fuzzy Time Series Forecasting Model. *Indian Journal of Science and Technology*, 11(27), 1-11. <https://doi.org/10.17485/ijst/2018/v11i27/130708>

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

### Take down policy

If you believe that this document breaches copyright please contact us at [vbn@aub.aau.dk](mailto:vbn@aub.aau.dk) providing details, and we will remove access to the work immediately and investigate your claim.

# A Weighted Fuzzy Time Series Forecasting Model

Daniel Ortiz-Arroyo and Jens Runi Poulsen

Department of Energy Technology, Aalborg University Esbjerg, Denmark;  
doa@et.aau.dk

## Abstract

In this paper we describe a new automatic partitioning method and a first order weighted fuzzy time series forecasting model. First, we show that our automatic fuzzy partitioning method provides an accurate approximation to the original time series. The fuzzy sets extracted from our partitioning are grouped to create a rule-base that will be used in forecasting. We found that the accuracy of our first order model is improved when an ordered weighting averaging operator is applied. The model presented in this paper does not attempt to produce the most accurate forecasting results, when compared with other more complex higher order models. Our goal is to show that there is still space for improvement when simple first order forecasting models are used. Our results show that the combination of a simple partitioning method, a first order model, and an averaging operator is still capable of outperforming not only the first order models that have proposed in the literature but also other higher order models.

**Keywords:** Aggregation Operators, Fuzzy Logic, Time Series Forecasting

## 1. Introduction

Forecasting is essential in decision making for domains such as financial markets, planning for energy production to satisfy future electricity demand and other economic and engineering applications. The current forecasting models include those based on optimization and statistics such as auto regression and hidden Markov chains and the methods based on computational intelligence techniques. Among these are artificial neural networks (ANN), support vector regression, evolutionary computing and fuzzy logic. The forecasting models vary in features, complexity, accuracy and applicability.

Among these methods ANN have shown very accurate results<sup>1</sup>.

This paper is about fuzzy time series forecasting. Forecasting models based on fuzzy logic, employ rules, relations, or operators<sup>2</sup> to create mathematical or com-

putational models capable of forecasting approximately future events, based on its past history. Fuzzy forecasting models have been applied in many domains such as: predicting university enrollments<sup>3-10</sup>, stock market indexes<sup>11</sup>, temperature prediction<sup>12</sup> and energy demand<sup>13</sup>.

The idea of using fuzzy time series in forecasting, was originally proposed by Song and Chissom<sup>3</sup>. The time-variant and time-invariant models described in their series works were tested on the problem of forecasting student enrollments at the University of Alabama.

Chen in<sup>5</sup> introduced a simplified procedure aimed at reducing the high computational overhead of his previous work and proposed a high order fuzzy time series model that showed significantly better results than its first order counterpart described in<sup>4</sup>.

The literature on fuzzy time series has recognized that some of the main factors that affect a model's performance are the number and length of the fuzzy intervals

\*Author for correspondence

and the content of the rules used. To address with these issues, Chen and Chung in<sup>7</sup> proposed a high order fuzzy time series model that uses genetic algorithms to optimize the fuzzy interval lengths. A similar procedure was proposed by Kuo et al. in<sup>9</sup>, using particle swarm optimization (PSO). A more recent research described in<sup>14</sup> employs also PSO to adjust and optimize the lengths of intervals in the universe of discourse.

More recently<sup>15</sup>, proposed to use clustering techniques to determine the optimal number and length of the intervals. In<sup>16</sup> a method based on intuitionist fuzzy sets is proposed to deal with the uncertainty in non-deterministic time series. All previous models require the use of more sophisticated rule production and selection methods, the use of optimization techniques or of extensive training and tuning to achieve the highest performance in prediction.

In this paper we propose a Weighted Fuzzy Time Series Forecasting model (*WFTSF*). First, we describe a simple automatic method to partition the time series into fuzzy intervals. Next, we discuss rule extraction and forecasting and explain how an Order Weighted Average (OWA) operator can improve our model.

Our experimental results show that in spite of its simplicity, our *WFTSF* model is capable of outperforming some of the more complex models reported in the literature.

Our model was tested using three datasets: the Alabama enrollment dataset<sup>4</sup> (with 22 entries), an energy demand dataset (with 12 entries) described in<sup>13</sup> and SBI, a stock market time series described in<sup>6</sup>. Our model was implemented using R programming language.

The rest of the paper is organized as follows. Section 2 provides an overall description of our proposed algorithm. Section 3 demonstrates how the proposed algorithm is used to extract and weight the forecasting rules. Section 4 presents experimental results. Finally, in section 5, we provide our conclusion.

## 2. Algorithm Overview

Our fuzzy time series model takes as input the time series datasets and provides as output the forecasts.

The algorithm is divided into two main components: fuzzification and rule forecasting.

The fuzzification component consists of six steps, where the first four steps are for data preprocessing and the last two comprise the fuzzification process itself. The ultimate goal of fuzzification is to generate a series of fuzzy sets or interval partitions and to establish associations between the fuzzy sets and the values in the dataset.

After the fuzzification process is completed, data is further processed by the rule forecasting component. In this stage, fuzzy sets are grouped into patterns and then transformed into forecasting rules. Finally, the rules are defuzzified and the forecasting is performed.

To compare forecasting performance with other models, we used the mean squared error (MSE) and mean absolute percentage error (MAPE). These measures are defined by the equations:

MSE	MAPE
$\frac{1}{n} \sum_{i=1}^n (F_i - A_i)^2$	$\frac{1}{n} \sum_{i=1}^n \frac{ F_i - A_i }{A_i} \times 100$

where  $A_i$  and  $F_i$  denote the actual output and forecast output at  $t_p$ , respectively.

The following sections explain in detail the fuzzification and rule forecasting processes performed by our algorithm.

## 3. Fuzzification and Forecasting of Enrollments

### 3.1 Fuzzifying Data

The proposed procedure for fuzzification can be described as the following six step process:

- Sort the values in ascending order.
- Compute the average distance between any two consecutive element values
- Eliminate outliers.
- Recalculate the new average distance between any two remaining consecutive values.

- Define the universe of discourse.
- Fuzzify the dataset.

First, the values in the historical dataset are sorted in ascending order to ease processing. Then the average distance between any two consecutive element values in the sorted dataset is computed. The average distance is obtained by the following equation:

$$AD(x_i, \dots, x_n) = \frac{1}{n-1} \sum_{i=1}^{n-1} |x_{p(i)} - x_{p(i+1)}|, \quad (1)$$

where  $p$  is a permutation that sorts the values in ascendant order i.e.  $x_{p(i)} \leq x_{p(i+1)}$ .

The standard deviation is computed as:

$$\sigma_{AD} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - AD)^2}. \quad (2)$$

Both the average distance and standard deviation are used in step 3 of our method to detect possible outlier values in the sorted dataset. The goal of this step is to eliminate these values from the calculation of the average distance. We defined arbitrarily an outlier as a value that is less than or larger than one standard deviation from the average i.e. all values not in  $value - \sigma \leq average \leq value + \sigma$ .

After the elimination of outliers, the average distance value is recalculated using the values remaining in the sorted dataset. This is the step 4 of the fuzzification process. The recalculated average distance is used in steps 5 and 6 to partition the universe of discourse into a series of trapezoidal fuzzy sets.

In step 3, the universe of discourse is defined by finding its lower and upper bounds, i.e. locating the largest and lowest values in the dataset. These values are subsequently adjusted by: (1) subtracting the average distance from the lowest value and (2) adding the average distance to the highest value.

More formally, if  $D_{max}$  and  $D_{min}$  are the highest and lowest values in the dataset, respectively, and  $AD$  is the

average distance, the universe of discourse,  $U$ , is defined as:

$$U = [D_{min} - AD, D_{max} + AD] \quad (3)$$

Once  $U$  has been determined, the fuzzy sets can be defined with symmetrical trapezoidal functions defined by points  $(a_1, a_2, a_3, a_4)$ . In symmetrical trapezoidal functions the distance between any pair of adjacent points in the core and support of the membership function has the same length e.g.  $a_2 - a_1 = a_3 - a_2 = a_4 - a_3$ .

The trapezoidal membership function  $\mu_A$  of a given fuzzy set  $A$  is obtained with the following equation:

$$\mu_A = \begin{cases} \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2 \\ 1, & a_2 \leq x \leq a_3 \\ \frac{a_4 - x}{a_4 - a_3}, & a_3 \leq x \leq a_4 \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

The range  $R$  is computed as:

$$R = UB - LB, \quad (5)$$

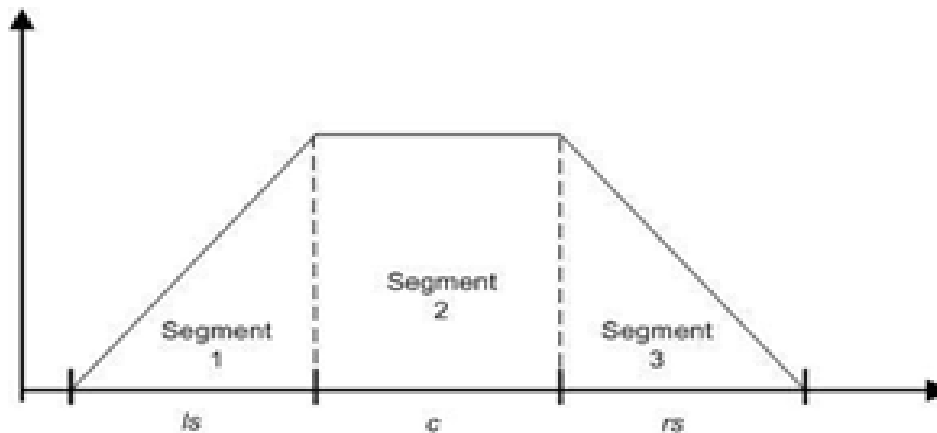
where  $UB$  and  $LB$  denote the upper bound and lower bound of  $U$  respectively.

All generated trapezoidal membership functions have the same segment lengths  $AD$ . Therefore, using this average distance  $ls = AD, c = AD$  and  $rs = AD$ , as is shown in Figure 1.

To calculate the number of fuzzy sets  $n$  defined over  $U$  we employ:

$$n = \frac{R - S}{2S} \quad (6)$$

The trapezoidal membership function procedure we have described is similar to the one in<sup>8</sup>. However, there are some important differences with our approach. For instance, in<sup>8</sup>, the number of sets is fixed and the segment length  $S$  is calculated solving equation 6 for  $S$ . Contrarily, in our approach we calculate the number of sets that will be needed in the problem with Equation 6 and use, as the segment length  $S$ , the recalculated average distance.



**Figure 1.** The segments of a trapezoidal fuzzy set.

Finally, instead of using the standard deviation to find the universe of discourse, as is done in<sup>8</sup>, we use the recalculated average distance as is shown in Equation 3.

### 3.2 Fuzzification Example

In the following example we will fuzzify the first four years (1971 - 1974) of student enrollment at the University of Alabama, to illustrate how the fuzzification process is performed by our method. The values to be fuzzified are 13055, 13563, 13867 and 14696.

First, since the sequence of these values is already in ascending order, the sorting step is omitted. The average distance and the standard deviation for this data set are calculated as:

$$AD = \frac{|13055-13563|+|13563-13867|+|13867-14696|}{3} = 547$$

and

$$\sigma_{AD} = \sqrt{\frac{(508-547)^2+(304-547)^2+(829-547)^2}{3}} \approx 216$$

Next, outliers i.e. values that are less than or larger than one standard deviation from  $AD$ , are discarded. In our example only values satisfying the condition:

$$547 - 216 \leq x \leq 547 + 216$$

are used to recalculate a new average distance. In this example, the only value that satisfies the above condition is 508. Thus the recomputed average distance  $AD_R$ , and the segment length  $S$ , are both set to 508. The idea of eliminating outliers is to avoid that the fuzzy partitions could be skewed by these values.

At this point, steps 1 - 3 of the fuzzification process have been completed. Now we determine the lower and the upper bounds of  $U$ . Using Equation 5 we obtain:

$$LB = 13055 - 508 = 12547$$

$$UB = 14696 + 508 = 15204$$

With these values we get  $U = [12547, 15204]$ . The range,  $R$ , is then computed as the difference between  $UB$

**Table 1.** Fuzzifying the first four years of enrollment

Fuzzy set	Trapezoidal fuzzy set	Crisp interval
$A_1$	(12547,13055,13602,14149)	$\mu_1 = [13055,13602]$
$A_2$	(13602,14149,14696,15402)	$\mu_1 = [14149,14696]$

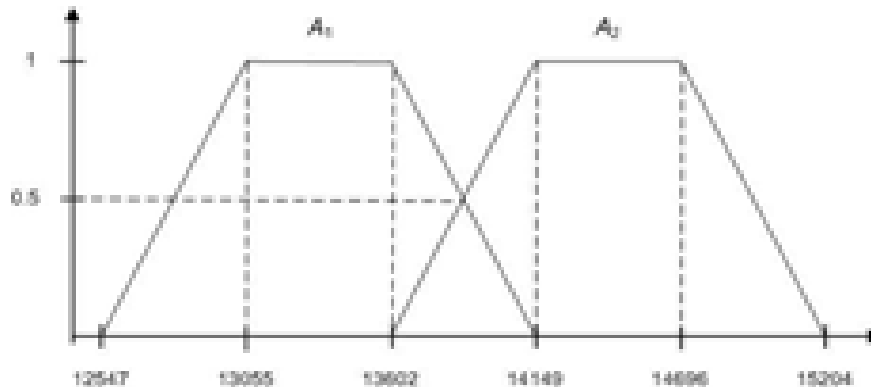


Figure 2. Generated membership functions.

and  $LB$  to get  $R = 15204 - 12547 = 2657$ . Finally the number of sets  $n$ , is computed as

$$n = \frac{2657 - 508}{2 \cdot 508} \approx 2$$

Once the universe of discourse and the parameters  $n$ ,  $R$  and  $S$  are determined, we generate the fuzzy sets as shown in Table 1 and Figure 2.

As shown in Table 1, points  $a_1$ ,  $a_2$ ,  $a_3$  and  $a_4$  are used to represent the trapezoidal fuzzy sets  $A_1$  and  $A_2$ . Note that the difference between the pairs of points  $|a_1 - a_2|$ ,  $|a_2 - a_3|$  and  $|a_3 - a_4|$ , representing the three segments of a trapezoidal fuzzy sets in Table 1 and Figure 2, is not exactly 508. This is because our algorithm adapts the segment length so that the lowest value in the dataset always appears as the left bound of the crisp interval, and the highest value in the dataset always appears as the right bound of the crisp interval.

From Table 1 and Figure 2 it can be seen that the lowest of the four values (i.e. 13055), appears as the lower bound of the first crisp interval  $u_1$ , and the highest value (i.e. 14696), appears as the upper bound in the second crisp interval,  $u_2$ . Generally, these values will not be matched due to rounding errors occurring as a result of applying Equations 1 and 6. Therefore our algorithm has to adjust slightly the segment lengths to fit into the universe of discourse.

Now we are able to fuzzify the first four historical enrollments according to membership functions  $A_1$  and  $A_2$ , defined by:

$$A_1 = \begin{cases} 0, & x < 12547 \\ \frac{x - 12547}{13055 - 12547}, & 12547 \leq x \leq 13055 \\ 1, & 13055 \leq x \leq 13602 \\ \frac{14149 - x}{14149 - 13602}, & 13602 \leq x \leq 14149 \\ 0, & x > 14149. \end{cases}$$

and

$$A_2 = \begin{cases} 0, & x < 13602 \\ \frac{x - 13602}{14149 - 13602}, & 13602 \leq x \leq 14149 \\ 1, & 14149 \leq x \leq 14696 \\ \frac{15204 - x}{15204 - 14696}, & 14696 \leq x \leq 15204 \\ 0, & x > 15204. \end{cases}$$

As Figure 3 indicates, the boundaries of  $A_1$  and  $A_2$  generally overlap. For instance, this happens in the case of the enrollment for year 1973, which was 13867. This element has a partial degree of membership in both sets  $A_1$  and  $A_2$ .

The membership degree of element 13867 in  $A_1$  is  $0.5155 \approx 0.52$ , and in  $A_2$ , it is  $0.4845 \approx 0.48$ . Hence the enrollment for 1973 is fuzzified as  $A_1$  since its membership degree is higher in this set. If the membership degree

was exactly 0.5, the element will be associated to either  $A_1$  or  $A_2$ .

The results of fuzzifying the first four years of enrollment in the University of Alabama are shown in Table 2.

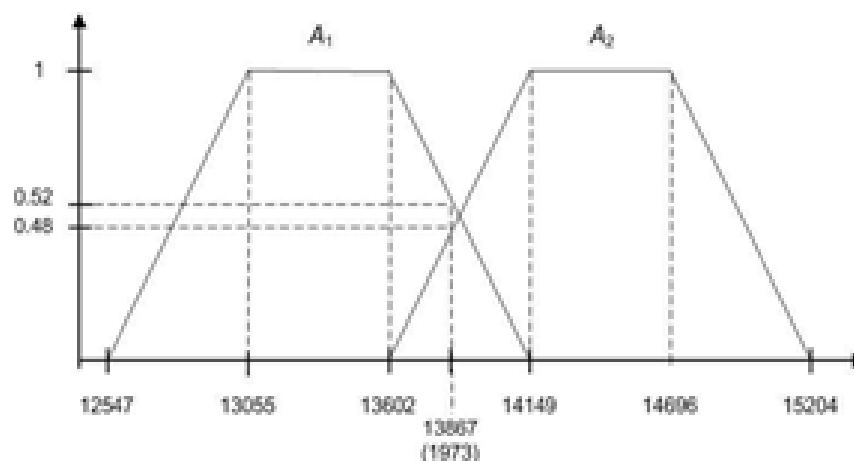
### 3.3 Fuzzifying the Dataset

Using the same procedure applied in the example from previous section, we derive the trapezoidal membership functions of the entire Alabama enrollment dataset from 1971 to 1992, shown in Table 3.

Similarly the fuzzified annual enrollments are listed in Table 4. It is important to remark that when assigning the fuzzy sets in Table 4, we have used *full knowledge* of the time series, to find the fuzzy set in which these values have maximum membership degree, as shown in Table 3.

From fuzzified data in Table 4, we get the first order relationships in Table 5.

Where  $A_j \rightarrow A_k$  may be interpreted as  $A_j$  being the current state at time  $t_{i-1}$  and  $A_k$  the next state at time  $t_i$  of a fuzzy time series. Using these *simple rules* with one antecedent and one consequent, we can approximate the fuzzy time series by defuzzifying the output. This is done by calculating the centroid of the fuzzy sets at time  $t_i$ . Calculating the centroid consists in finding the midpoint  $m_i$  of the core interval in the membership function  $u_i$ , given by  $m_i = (a_2 + a_3)/2$ , since the membership functions are symmetrical trapezoids defined by  $A_i = (a_1, a_2, a_3, a_4)$ . Our model obtained MAPE=9609.4 and MSE=0.50%. These results represent the optimal maximum forecasting accuracy that we can expect to obtain with our proposed partitioning method.



**Figure 3.** Membership functions generated.

**Table 2.** Fuzzified Enrollments 1971 - 1974.

Year	Enrollment	Fuzzy set	Membership degree
1971	13055	$A_1$	1
1972	13563	$A_1$	1
1973	13867	$A_1$	0.52
1974	14696	$A_2$	1



### 3.4 Rule Forecasting

To perform forecasting we need to find *composed rules* that will solve the ambiguities in the rule base. This is done by grouping all rules that have the same antecedent

or current state. After this grouping we get the following simple and composed rules in Table 3:

To perform forecasting, we use the process suggested in<sup>4</sup> consisting in applying two basic principles.

**Principle 1:** For matching single rules:

$$A_i \rightarrow A_j$$

where  $A_i$  is the fuzzified number for year  $i$ . We forecast a value for time  $t_i+1$  associated with fuzzy set  $A_j$  by calculating the centroid or midpoint  $m_j$  in fuzzy number  $A_j$ .

**Principle 2:** For matching composed rules:

$$A_i \rightarrow A_p A_k \dots A_m$$

where  $A_i$  is the fuzzified number for year  $i$ . This means that the forecasting value for time  $t_i+1$  will be calculated as the aggregated value of the midpoints of the fuzzy sets  $\{j,k,\dots,m\}$ .

To apply these principles we need to *train* our model to extract the fuzzy sets and rules in the time series as was described in section 3.1.

Forecasting for time  $t_i+1$  consist in finding the rule that has fuzzy set  $A_i$  associated to time  $t_i$  as its antecedent. Once the rule is found we apply Principles 1 or 2. It must be noted that once a forecast for time  $t_i+1$  is produced and the real value is known, the time series is updated to include the new value. The updated time series is then fuzzified again using the six fuzzification steps to create a new rule base. The new rule base is then used to forecast a value for time  $t_{i+2}$ . This process continues for the rest of forecasts.

Forecasting with composed rules, is done by calculating the average of the centroids in the consequent of the rules. This works well when the time series changes smoothly i.e. when the difference between the centroids used in the composed rules is small. However, if the difference is large, the average may not be a good representation for all the values in the antecedent. Additionally, if the same rule is applied at several time intervals we will get repeated forecasting values. To add flexibility in forecasting, we need a way to change our forecasts according to how close the centroids in the consequent of a rule are. For this purpose we may use an OWA operator<sup>17</sup>.

**Table 3.** Generated fuzzy sets.

Fuzzy Set	Trapezoid Function
$A_1$	(12861,13055,13245,13436)
$A_2$	(13245,13436,13626,13816)
$A_3$	(13626,13816,14007,14197)
$A_4$	(14007,14197,14388,14578)
$A_5$	(14388,14578,14768,14959)
$A_6$	(14768,14959,15149,15339)
$A_7$	(15149,15339,15530,15720)
$A_8$	(15530,15720,15910,16101)
$A_9$	(15910,16101,16291,16482)
$A_{10}$	(16291,16482,16672,16862)
$A_{11}$	(16672,16862,17053,17243)
$A_{12}$	(17053,17243,17433,17624)
$A_{13}$	(17433, 17624,17814,18004)
$A_{14}$	(17814, 18004,18195,18385)
$A_{15}$	(18195,18385,18576,18766)
$A_{16}$	(18576,18766,18956,19147)
$A_{17}$	(18956,19147,19337,19531)



**Table 4.** Fuzzified Annual Enrollments

Year	Enrollment	Fuzzy Set
1971	13055	$A_1$
1972	13563	$A_2$
1973	13867	$A_3$
1974	14696	$A_5$
1975	15460	$A_7$
1976	15311	$A_7$
1977	15603	$A_7$
1978	15861	$A_8$
1979	16807	$A_{11}$
1980	16919	$A_{11}$
1981	16388	$A_{10}$
1982	15433	$A_7$
1983	15497	$A_7$
1984	15145	$A_6$
1985	15163	$A_6$
1986	15984	$A_8$
1987	16859	$A_{11}$
1988	18150	$A_{14}$
1989	18970	$A_{16}$
1990	19328	$A_{17}$
1991	19337	$A_{17}$
1992	18876	$A_{16}$

The OWA operator is a flexible operator that is capable of producing a wide range of values from the minimum to the maximum in the set of aggregated values, including the average. The value produced depends on the weights assigned to the OWA operator. Our model changes the weights in the OWA operator to produce different *orness*

values<sup>17</sup>. The idea is to use these *orness* values to produce a better approximation to the time series when certain conditions occur. The OWA operator with *orness*3/4 is used when there is a *large* difference in the centroids of the consequent in a fuzzy rule. If the difference in centroids is *small* the best approximation is to calculate the aver-

**Table 5.** First order relations extracted from the time series

$A_1 \rightarrow A_2$	$A_8 \rightarrow A_{11}$	$A_6 \rightarrow A_8$
$A_2 \rightarrow A_3$	$A_{11} \rightarrow A_{11}$	$A_{11} \rightarrow A_{14}$
$A_3 \rightarrow A_5$	$A_{11} \rightarrow A_{10}$	$A_{14} \rightarrow A_{16}$
$A_5 \rightarrow A_7$	$A_{10} \rightarrow A_7$	$A_{16} \rightarrow A_{17}$
$A_7 \rightarrow A_7$	$A_7 \rightarrow A_6$	$A_{17} \rightarrow A_{17}$
$A_7 \rightarrow A_8$	$A_6 \rightarrow A_6$	$A_{17} \rightarrow A_{16}$

**Table 6.** Disambiguated rule base.

$G1 : A_1 \rightarrow A_2$	$G7 : A_{11} \rightarrow A_{10}, A_{11}, A_{14}$
$G2 : A_2 \rightarrow A_3$	$G8 : A_{10} \rightarrow A_7$
$G3 : A_3 \rightarrow A_5$	$G9 : A_6 \rightarrow A_6, A_8$
$G4 : A_5 \rightarrow A_7$	$G10 : A_{14} \rightarrow A_{16}$
$G5 : A_7 \rightarrow A_6, A_7, A_8$	$G11 : A_{16} \rightarrow A_{17}$
$G6 : A_8 \rightarrow A_{11}$	$G12 : A_{17} \rightarrow A_{16}, A_{17}$

age using an *orness* value of 1/2. These *orness* values are aimed at moving the forecasts closer to the largest centroid value. The *orness* of 3/4, was chosen

since its associated weights in the OWA operator can be easily calculated using an analytical closed form described by<sup>18</sup>, with the following equation, for Orness=3/4:

$$w_i = \frac{1}{n} \sum_{j=i}^n \frac{1}{j} \quad (7)$$

Our model applies the following rule to decide which orness value should be used in a composed rule such as  $A_j \rightarrow A_k A_i$ :

$$g_{centroid} = \begin{cases} \text{if } |centroid_k - centroid_l| \geq R - 2 \cdot avg \\ \quad \text{apply OWA with orness} = 3/4 \\ \text{otherwise apply OWA with orness} = 1/2 \end{cases}$$

where *gcentroid* is the resulting centroid, *R* is the range of the time series as defined by Equation 5 and *avg* is the average distance or segment length of the trapezoidal functions. We call our automatic partitioning method together with the application of the principles 1 and 2, and the OWA operator our *WFTSF* model.

## 4. Algorithm Evaluation

Table 7 shows a comparison of the results obtained by WFTSF with other higher order models that have been optimized. The table shows that WFTSF model outperforms the first order models in<sup>4,8</sup> but also the higher order models in<sup>5,19,20</sup> in terms of MAPE and MSE. The rest of

**Table 7.** Results of different first and higher order models

	<b>Chen[4] order=1</b>	<b>Cheng[8] order=1</b>	<b>Chen [5] order=3</b>	<b>Li [19] order=11</b>	<b>Sing [20] order=3</b>	<b>Kuo [9] order=9</b>	<b>WFTSF order=1</b>
MSE	407507	261162	86694	85040	76509	234	84005
MAPE	3.11	2.66	1.53	1.53	1.41	0.014	1.09

the higher order algorithms perform better than WFTSF due to the use of higher order models and optimization techniques.

We also applied our model to the time series of electricity demand reported in<sup>13</sup>, where our model obtained  $MAPE = 68563$  and  $MSE = 4.77$ , which shows that with half of the data we used in the University of Alabama dataset, the model increases MAPE significantly. Finally, we tested our model with the time series of the daily stock exchange price of SBI reported in<sup>6</sup>. Our model obtained  $MSE = 1255.46$  and  $MAPE = 0.90$  whereas the more complex method described in<sup>6</sup> obtained a significantly better result with  $MSE = 782.46$ .

## 5. Conclusions

In this paper we have presented a new and simple first order fuzzy time series forecasting model that generates automatically a partition of the universe of discourse in intervals, using the variations observed in the datasets. We also show that its forecasting accuracy can be improved with an OWA operator. Our experiments show that *WFTSF*, outperforms in forecasting accuracy not only the results obtained by other first order models, but also some higher order models in terms of MAPE and MSE. Our future work includes designing a more accurate and higher order model based on optimizing the weights applied to the rules.

## 6. References

1. Ortiz-Arroyo D, Skov MK, Huynh Q. Accurate electricity load forecasting with artificial neural networks. Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA '05. IEEE Computer Society. 2005; 1:94–9.
2. Klir GJ, Yuan B. Fuzzy sets and fuzzy logic: theory and applications. Prentice Hall. 1995.
3. Song Q, Chissom BS. Forecasting enrollments with fuzzy time series Part I. Fuzzy Sets and Systems. 1993; 54(1):1–9. Crossref.
4. Chen S-M. Forecasting enrollments based on fuzzy time series. Fuzzy Sets and Systems. 1996; 81(3):311–9.
5. Chen S-M. Forecasting enrollments based on high-order fuzzy time series. Cybernetics and Systems. 2002; 33(1):1–16. Crossref.
6. Singh S. A robust method of forecasting based on fuzzy time series. Applied Mathematics and Computation. 2007; 188(1):472–84. Crossref.
7. Chen S-M, Chung N-Y. Forecasting enrollments using high-order fuzzy time series and genetic algorithms. International Journal of Intelligent Systems. 2006; 21(5):485–501. Crossref.
8. Cheng C-H, Chang J-R, Yeh C-A. Entropy-based and trapezoid fuzzification based fuzzy time series approaches for forecasting it project cost. Technological Forecasting and Social Change. 2006; 73(5):524:42. Crossref.
9. Kuo I-H, Horng S-J, Kao T-W, Lin T-L, Lee C-L, Pan Y. An improved method for forecasting enrollments based on fuzzy time series and particle swarm optimization. Expert Systems with Applications. 2009; 36(3, Part 2):6108–17. Crossref.
10. Stevenson M, Porter JE. Fuzzy time series forecasting using percentage change as the universe of discourse. International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering. 2009; 3(7):464–7.
11. Chen T-L, Cheng C-H, Teoh H-J. High-order fuzzy time-series based on multiperiod adaptation model for forecasting stock markets. Physica A: Statistical Mechanics and Its Applications. 2008; 387(4):876–88. Crossref.
12. Chen S-M, Hwang J-R. Temperature prediction using fuzzy time series. IEEE Transactions on Systems, Man, and Cybernetics, Part B. 2000; 30(2):263–75. Crossref. PMid:18244753

13. Mirasgedis S, Sarafidis Y, Georgopoulou E, Lalas D, Moschovits M, Karagiannis F, Papakonstantinou D. Models for mid-term electricity demand forecasting incorporating weather influences. *Energy* 2006; 31(23):208–27. Crossref.
14. Huang Y-L, Horng S-J, He M, Fan P, Kao T-W, Khan MK, Lai J-L, Kuo I-H. A hybrid forecasting model for enrollments based on aggregated fuzzy time series and particle swarm optimization. *Expert Systems with Applications*. 2011; 38(7):8014–23. Crossref.
15. Chen S-M, Wang N-Y, Pan J-S. Forecasting enrollments using automatic clustering techniques and fuzzy logical relationships. *Expert Systems with Applications*. 2009; 36(8):11070–6. Crossref.
16. Kumar S, Gangwar SS. Intuitionistic fuzzy time series: An approach for handling nondeterminism in time series forecasting. *IEEE Transactions on Fuzzy Systems*. 2016; 24(6):1270–81. Crossref.
17. Yager RR. On ordered weighted averaging aggregation operators in multi criteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*. 1988; 18(1):183–90. Crossref.
18. Ahn BS. On the properties of OWA operator weights functions with constant level of orness. *IEEE Transactions on Fuzzy Systems*. 2006; 14(4):511–5. Crossref.
19. Li S-T, Cheng Y-C. Deterministic fuzzy time series model for forecasting enrollments. *Computers & Mathematics with Applications*. 2007; 53(12):1904–20. Crossref.
20. Singh SR. A simple time variant method for fuzzy time series forecasting. *Cybernetics and Systems*. 2007; 38(3):305–21. Crossref.